Software Requirements Specification

for

TaskEase

Version 0.1 Beta approved

Prepared by Alessio Sordo

University of Ferrara

19.11.2023

Table of contents

Revision History

Name	Date	Reason For Changes	Version
alesordo	21.09.23	First draft	0.1 Alpha
alesordo	27.09.23	Introduction completed	0.1 Alpha-2
alesordo	29.09.23	Overall description completed	0.1 Alpha-3
alesordo	03.10.23	Completed Chapter 3, apart from 3.1	0.1 Alpha-4
alesordo	05.10.23	Completed Chapter 3	0.1 Alpha-4
alesordo	23.10.23	Completed Chapters 4 and 5, missing ER diagram and REST API specification	0.1 Alpha-5
alesordo	27.10.23	Added ER diagram – added section 4.7	0.1 Alpha-6
alesordo	19.11.23	Edited some major flaws. Missing REST API specification	0.1 Beta

1. Introduction

1.1 Purpose

The purpose of this document is to present the detailed description of TaskEase, a web task manager application. It will explain the purpose and features of the software, its interfaces and what the software will do.

1.2 Document Conventions

This document will be referred to as SRS. It was created based on the IEEE template for System Requirements Specification Documents.

1.3 Intended Audience and Reading Suggestions

This document is addressed to:

- Day-to-day users, such as students, who want to use TaskEase for tracking their tasks efficiently. They may be interested particularly in section 3.1;
- Advanced users, such as professionals, who are interested in using TaskEase for more demanding task-tracking operations. All the functions are listed in section 2.2;
- Programmers, who want to further develop TaskEase.

1.4 Product Scope

TaskEase is a web service that allows users to manage their tasks efficiently. They can create tasks, assign them to specific users, mark their status as to do, in progress or done and document their progress. The program is based on Kanban, a project management system originally developed for manufacturing tasks. The main goal of TaskEase is to provide users with an easy-to-use service, with a clean UI and understandable functionalities, following a “less is more” philosophy.

1.5 References

- TaskEase GitHub: alesordo/TaskEase: Task management web service in a Kanban way (github.com);
- Kanban guide: https://kanbanguides.org/;

- IEEE Template for SRS: https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc;
- MIT license: The MIT License – Open Source Initiative;
- Priority components explanation in project management: http://www.bawiki.com/wiki/Matrix-Prioritization.html.

2. Overall Description

2.1 Product Perspective

TaskEase was developed for everybody that needs to manage tasks and projects. It can handle collaboration between people and a simple deployment on every server through Docker. The tasks follow a Kanban “Three Bins” structure, to visualize them immediately and have a visual feedback on the progress.

It’s an open-source project, open to any kind of improvement. It can run in every device that has a browser and an internet connection. It also works on mobile devices, even though the interface is still not optimized for them.

2.2 Product Functions

Authentication page:

- User registration;
- User login.

Projects page:

- List projects;
- Create project;
- Delete project.

Project detail page:

- Add collaborator;
- Edit project details.

Project page:

- Create tasks → details:
    - Assign title;
    - Estimate task time;
    - Assign to user;
    - Set due date;
    - Add description.
- List tasks in a three-bins fashion;

- Edit tasks status by dropping them into a specific bin;
- Filter tasks;

Task page:

- Update task details;
- Change task status;
- Document task progress and elapsed time;
- Edit comments.

2.3 User Classes and Characteristics

- Day-to-day users, such as students, who want to use TaskEase for tracking their tasks efficiently. They are the primary recipients of the basic functionalities of this product;
- Advanced users, such as professionals, who are interested in using TaskEase for more demanding task-tracking operations. They may take advantage of the collaboration function of the product, to manage complex projects in group;
- Programmers, who want to further develop TaskEase. They will use the source files, which can be found on the GitHub repository.

2.4 Operating Environment

Containerized TaskEase can be executed on every server that runs Docker (like Windows, MacOS or Linux). Its source code can also run on every modern machine, even though some dependencies must be installed autonomously.

2.5 Design and Implementation Constraints

TaskEase is developed following a Three-tier architecture. In fact, it’s composed of a MongoDB Database for persisting data, a back-end written in JavaScript using the Express NodeJs framework and a front-end written in TypeScript using the Angular framework. These components are binded together and all needed for the correct functioning of the product. TaskEase is provided as-it-is and its maintenance won’t be guaranteed by the developer.

2.6 User Documentation

A basic user guide of TaskEase can be found on the README of the GitHub repository.

2.7 Assumptions and Dependencies

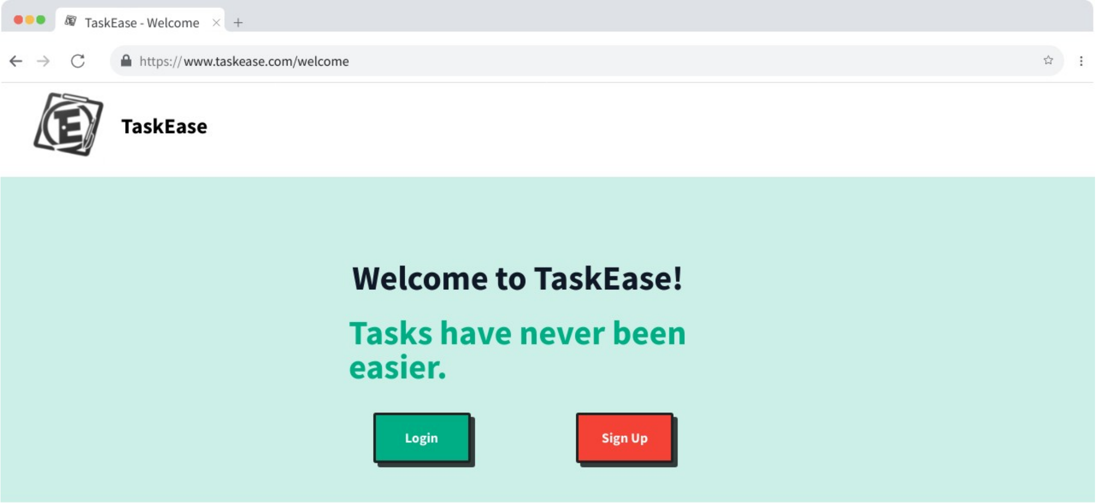
TaskEase, if deployed using Docker, shouldn’t need any installed component on any system to run. In fact, Docker will manage all the dependencies needed to build the project. However, building the project from the source code could require the user to install some dependencies, namely: Node.js, npm, Express, Angular, MongoDB. In this case users should check by themselves the needed steps to install the missing components.

3. External Interface Requirements

3.1 User Interfaces

The following are user interfaces mock-ups. The real user interfaces may differ from these concepts.

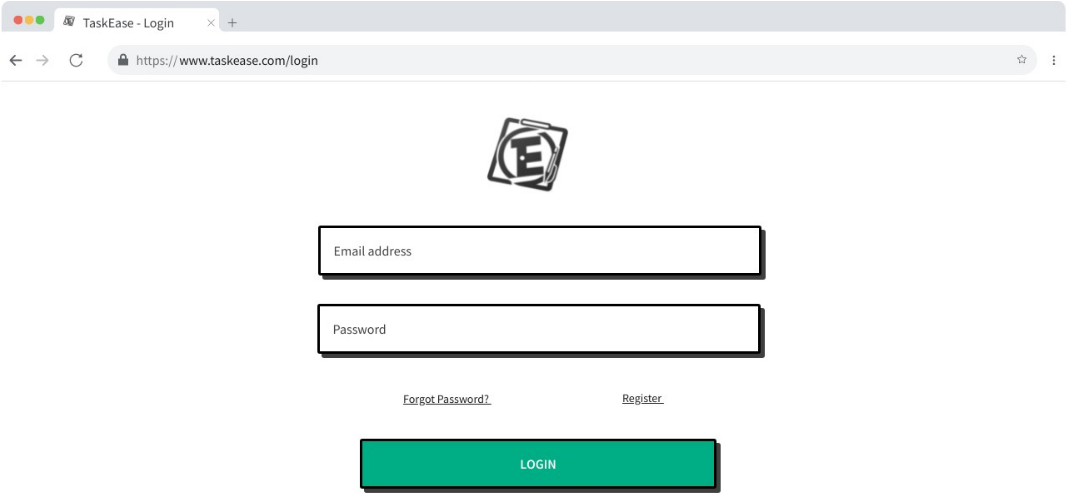
1. Landing page when user isn't authenticated:



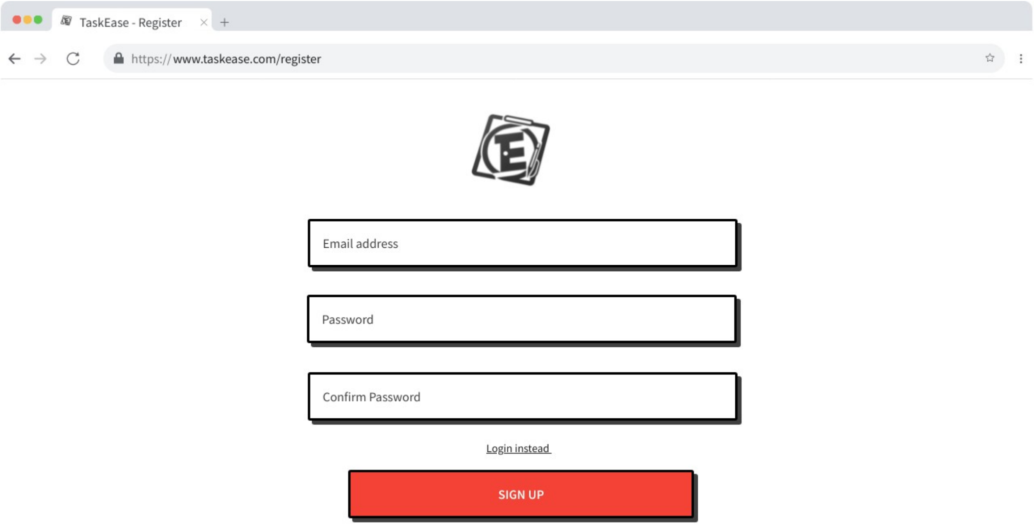
Welcome to TaskEase!

Tasks have never been easier.

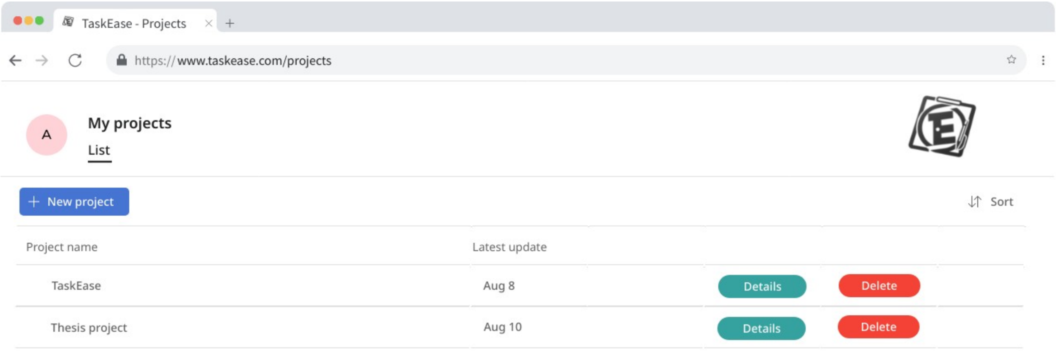
2. Login page:



3. Registration page:

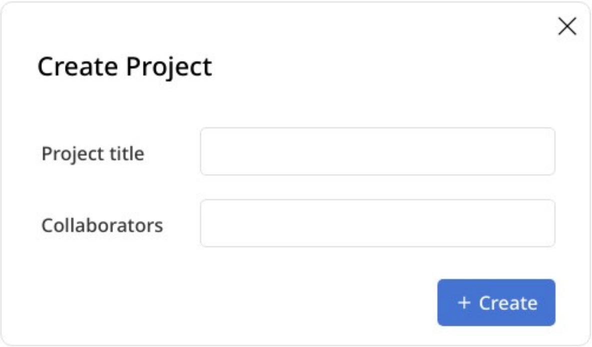


4. List of all projects, used as landing page when user is authenticated:

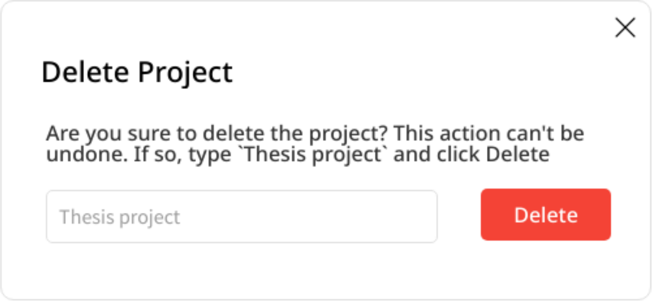


Project name	Latest update		
TaskEase	Aug 8	Details	Delete
Thesis project	Aug 10	Details	Delete

5. Project creation dialog, shown when clicking “New project" on interface 4:



6. Project deletion dialog, shown when clicking “Delete” on interface 4:

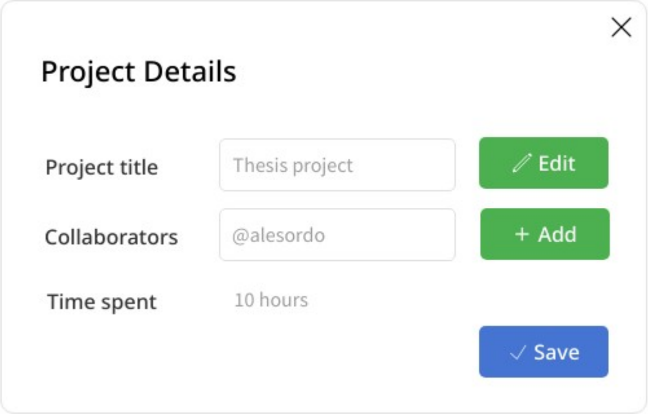


Delete Project

Are you sure to delete the project? This action can't be undone. If so, type `Thesis project` and click Delete

Thesis project

7. Project details dialog, shown when clicking “Details” on interface 4:



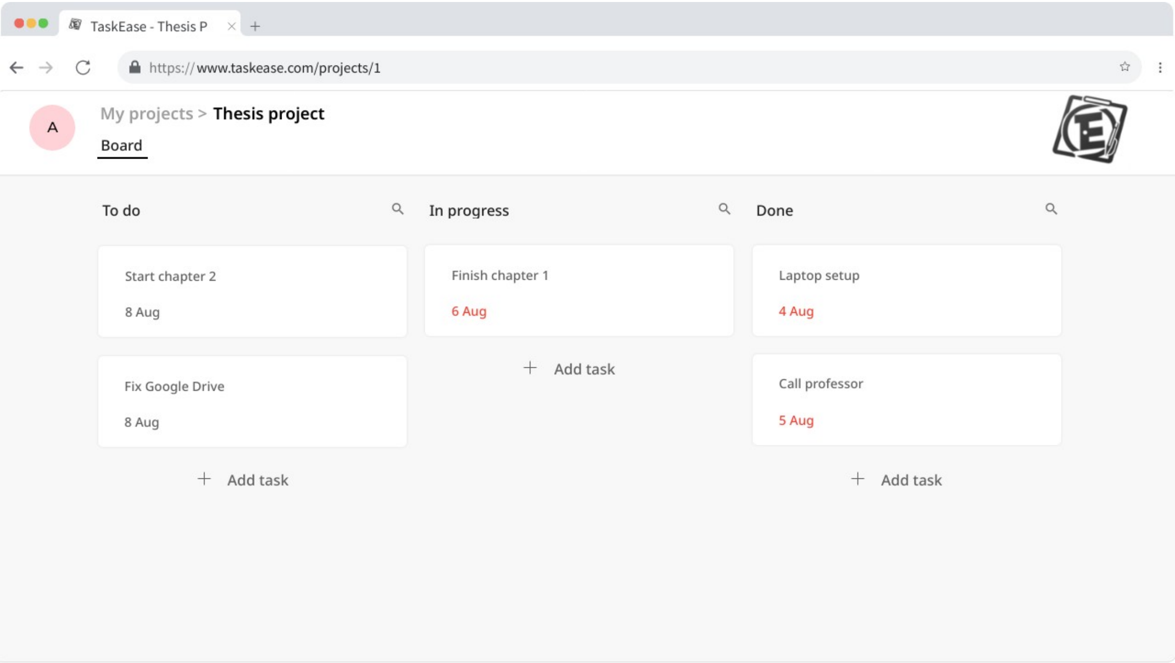
Project Details

Project title: Thesis project

Collaborators: @alesordo

Time spent: 10 hours

8. Project Kanban board. Dragging a task from a bin to another changes its status:



TaskEase - Thesis P

https://www.taskease.com/projects/1

My projects > Thesis project

Board

To do

Start chapter 2
8 Aug

Fix Google Drive
8 Aug

In progress

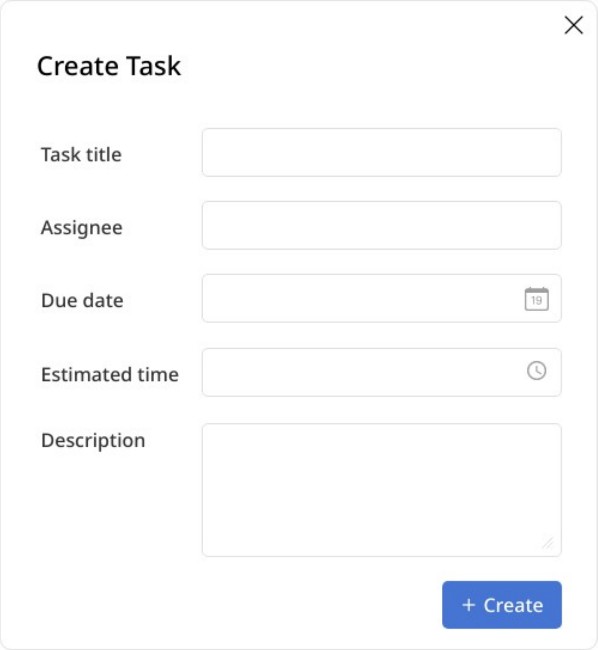
Finish chapter 1
6 Aug

Done

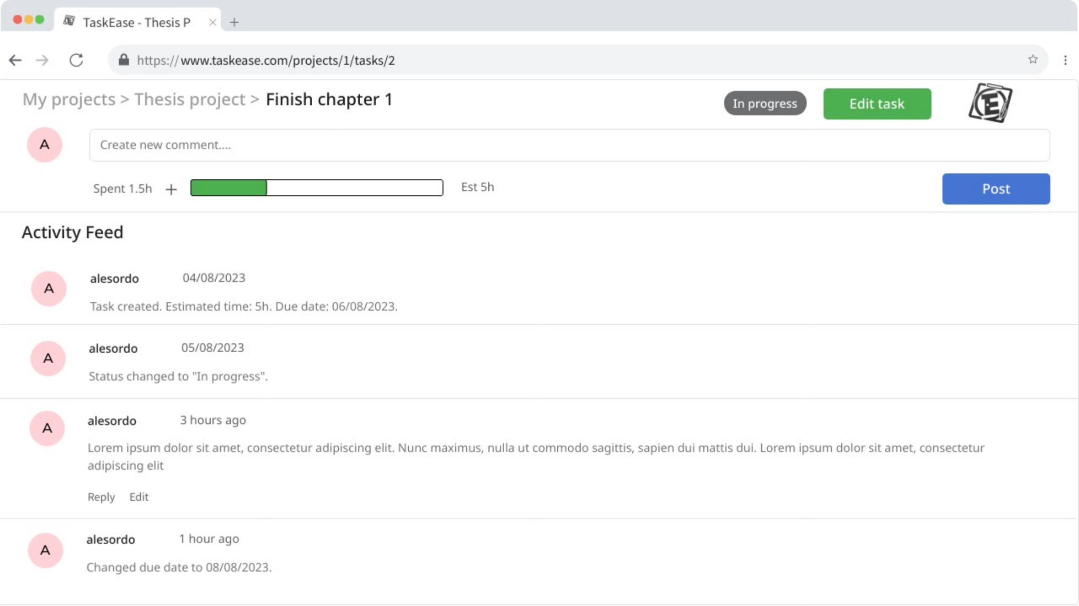
Laptop setup
4 Aug

Call professor
5 Aug

9. Task creation dialog, shown when clicking “Add task” on interface 8:



10. Task page. Comments can be created by typing in the comment box and clicking “Post”. Users can reply to comments and edit them:



A alesordo 04/08/2023

Task created. Estimated time: 5h. Due date: 06/08/2023.

A alesordo 05/08/2023

Status changed to "In progress".

A alesordo 3 hours ago

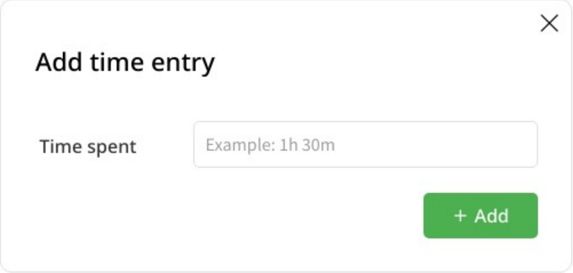
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nunc maximus, nulla ut commodo sagittis, sapien dui mattis dui. Lorem ipsum dolor sit amet, consectetur adipiscing elit

Reply Edit

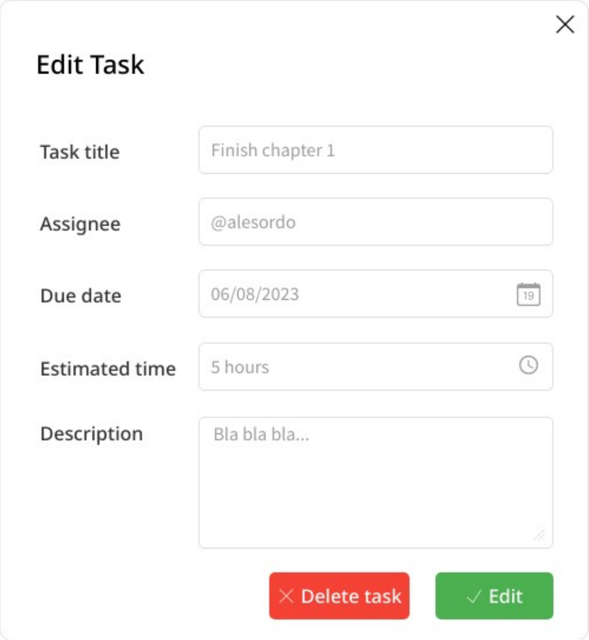
A alesordo 1 hour ago

Changed due date to 08/08/2023.

11. Add time dialog, shown when clicking “+” on interface 10:



12. Edit task dialog, shown when clicking “Edit task” on interface 10:



Edit Task

Task title: Finish chapter 1

Assignee: @alesordo

Due date: 06/08/2023

Estimated time: 5 hours

Description: Bla bla bla...

3.2 Hardware Interfaces

The only hardware required to run a TaskEase instance is a machine equipped with either Windows, Linux or Mac OS operating systems; cross-compatibility is guaranteed by Docker. No additional hardware is needed.

3.3 Software Interfaces

TaskEase, as described in 2.5, is connected with some software components.

The first of them is the MongoDB database, that saves all the data of a deployment. The Entity-Relationship Diagram of the DB can be found in section Appendix A: Analysis Models.

The tables representing entities and relationships are created directly by the mongoose library via the so called “Data models”. Mongoose built-in functions are then used to perform DB CRUD operations.

Express manages the business logic of the application exposing operations as REST endpoints. The REST API of TaskEase consists of these URLs:

Endpoint	HTTP method	Actions	Status code
/			

Finally, the REST API endpoints are used to create the front-end in Angular.

The specific software components that interact in the context of the product are the following, including their version used for development and testing:

- MongoDB 7.0;
- Node.js 20.8.0;
- npm 10.1.0;
- Express 4.18.2;
- Angular 16.x.

3.4 Communications Interfaces

TaskEase can be deployed either locally or online. In both cases, a web browser is needed to run the application and use it.

If deployed online, an internet connection is required and you may need to set up a Web Server like Nginx to easily manage the application. An internet connection is also needed to update all the software components described in the above section 3.3.

4. System Features

Below are described the functional requirements of TaskEase. They are organized by features, the major services provided. Priority components are ranked on a scale from 1 to 9.

4.1 User authentication

4.1.1 Description and Priority

The feature for accessing all TaskEase functionalities. It requires users to be previously registered.

It has a high priority, being it essential for authorizing users to access their personal projects and tasks.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 2;
- Risk: 2.

4.1.2 Stimulus/Response Sequences

If users are not yet authenticated, a landing page will be displayed, offering the option to “Login”. Clicking this option will show the login page. By entering email and password and confirming, users will authenticate on TaskEase. If authentication is successful, the personal projects page will be displayed.

4.1.3 Functional Requirements

REQ-1: Authentication page must require email and password;
REQ-2: An error must be returned if the email + password combination doesn’t exist;
REQ-3: The login form mustn’t be sent if the email value is invalid;
REQ-4: Users have to be redirected to their projects page when authentication is successful.

4.2 User registration

4.2.1 Description and Priority

This feature allows new users to get access to TaskEase.

It has a high priority, as, together with the authentication feature, it allows users to use TaskEase functionalities.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 3;
- Risk: 3.

4.2.2 Stimulus/Response Sequences

If users are not yet registered, a landing page will be displayed, offering the option to “Sign up”. Clicking this option will show the registration page. By entering a username, an email, a password twice and confirming, users will register on TaskEase. If the registration process was successful, they will receive a confirmation email. By executing the action required, they’ll be successfully registered.

4.2.3 Functional Requirements

REQ-1: Authentication page must require username, email, password and password confirmation;
REQ-2: An error must be returned if the email and/or username already exist;
REQ-3: The registration form mustn’t be sent if either the email or password format is invalid;
REQ-4: Users have to receive a confirmation email after submitting the registration form successfully;
REQ-5: The registration will complete only after clicking a link or button on the confirmation email;
REQ-6: After confirming, users must be redirected to the login page, but only if another user is not yet authenticated.

4.3 Projects overview

4.3.1 Description and Priority

This feature displays all the projects the logged-in user is working or collaborating on.

It has a high priority, as project listing is a basic feature of every Kanban software, since each board is associated with a project and the lists of project is the way to access a specific board.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 2;
- Risk: 3.

4.3.2 Stimulus/Response Sequences

If users go to TaskEase root URL and are already logged in, they will be redirected to their list of projects.

4.3.3 Functional Requirements

REQ-1: All projects created by users or on which users have collaborated must be listed;
REQ-2: There must be at least a sorting filter, to easily search for a project among many;
REQ-3: Clicking on a specific project must redirect to its Kanban board.

4.4 Project creation

4.4.1 Description and Priority

This feature allows users to create new projects.

It has a high priority as creating a project is the mandatory step to generate a Kanban board.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 4;
- Risk: 3.

4.4.2 Stimulus/Response Sequences

On the page listing all projects described in section 4.3 if users click the "New project" button, a dialog will open, in which they can specify a project's name and, eventually, collaborators. If the project name isn't already present in their projects overview, pressing "Create" will successfully create a new project.

4.4.3 Functional Requirements

REQ-1: When creating a project, users must be asked to enter a title and possibly the username of one or more collaborators;
REQ-2: If a project with the same name already exists for the logged-in user, an error must be returned;
REQ-3: The logged-in user who creates the project is automatically a collaborator of it.

4.5 Project editing

4.5.1 Description and Priority

This feature allows users to edit their projects.

It has a medium priority, since project creation and listing are more basic functionalities of this product.

Priority components:

- Benefit: 7;
- Penalty: 9;
- Cost: 1;
- Risk: 2.

4.5.2 Stimulus/Response Sequences

On the page listing all projects described in section 4.3 if users click the “Details” button, a dialog will open, in which it’s possible to edit the project’s name and collaborators. After editing, the changes can be saved by clicking “Save”.

4.3.3 Functional Requirements

REQ-1: If a project with the same name already exists for the logged-in user, an error must be returned;
REQ-2: Collaborators can either be removed or added, including the account, excluding the logged-in account;
REQ-3: At least one collaborator must be kept on the project, an error is triggered otherwise.

4.6 Project deletion

4.5.1 Description and Priority

This feature allows users to delete their projects.

It has a medium priority, since project creation and listing are more basic functionalities of this product.

Priority components:

- Benefit: 7;
- Penalty: 9;
- Cost: 1;
- Risk: 2.

4.5.2 Stimulus/Response Sequences

On the page listing all projects described in section 4.3 if users click the “Delete” button, a dialog will open, in which it’s possible to delete the project. To proceed, the user must enter the project’s correct name.

4.3.3 Functional Requirements

REQ-1: The name of the project must be typed correctly by the user, if so the project is deleted forever, otherwise a warning message shows up.

4.7 Project logged-in user unassignment

4.5.1 Description and Priority

This feature allows logged-in users to unassign themselves from a project, without deleting it.

It has a medium priority, since project creation and listing are more basic functionalities of this product.

Priority components:

- Benefit: 6;
- Penalty: 8;
- Cost: 1;
- Risk: 2.

4.5.2 Stimulus/Response Sequences

On the page listing all projects described in section 4.3 if users click the “Exit project” button, a dialog will open, in which it’s possible to unassign themselves from the project. To proceed, the user must confirm in an additional dialog.

4.3.3 Functional Requirements

REQ-1: When unassigning the project, a dialog must pop-up to ask for a confirmation.

4.8 Kanban project board

4.5.1 Description and Priority

This feature lists all the tasks associated with a specific project on a Kanban board.

It has a high priority, as Kanban boards are the main feature of TaskEase.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 3;
- Risk: 3.

4.5.2 Stimulus/Response Sequences

After clicking on a specific project, users are redirected to its Kanban board with all the tasks associated.

4.3.3 Functional Requirements

REQ-1: Each project must be displayed on a specific bin based on its status of the moment;
REQ-2: The Kanban bins must be "To do", "In progress" and "Done";
REQ-3: Projects are listed in each bin based on the time when their status was set, from most to least recent;
REQ-4: There must be a search feature on each bin, in case the tasks become too much to search them quickly.

4.9 Task creation

4.5.1 Description and Priority

This feature allows to create new tasks of a project.

It has a high priority, since creating tasks is fundamental for populating the Kanban board.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 4;
- Risk: 4.

4.5.2 Stimulus/Response Sequences

On the Kanban board, if users click on "Add task" they will be prompted with a dialog to create a new task. After compiling all the fields and clicking "Create", if they're all valid, the task is created.

4.3.3 Functional Requirements

REQ-1: When creating a new task, users must specify: its title, its assignee (only one user), its time estimate and the due date;
REQ-2: A description of the task can be provided, but it's not mandatory;
REQ-3: If the value of one or more fields is invalid, an error must be prompted and the task isn't created;
REQ-4: Creating a task from the button of a specific bin on the Kanban board sets its status to the one of the bin automatically.

4.10 Task overview

4.5.1 Description and Priority

This feature allows to see a specific task's information, including its status, the elapsed time and comments posted.

It has a high priority, since tasks details are useful for documenting each step of a project.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 4;
- Risk: 4.

4.5.2 Stimulus/Response Sequences

On the Kanban board described in section 4.8, if users click on a specific task, its overview will open.

4.3.3 Functional Requirements

REQ-1: The overview must show the activity occurred on the task (activity feed), the elapsed time, the due date and the status;
REQ-2: On the activity feed all activities must be listed from least to most recent;
REQ-3: The activity feed must include comments and changes on the task's status (due date, elapsed time, title changes, assignee modification).

4.11 Task commenting

4.5.1 Description and Priority

This feature allows to add comments on a specific task.

It has a high priority, as comments give just more granularity to the task's status, but it could be difficult to implement.

Priority components:

- Benefit: 7;
- Penalty: 9;
- Cost: 4;
- Risk: 3.

4.5.2 Stimulus/Response Sequences

On the task overview described in section 4.10, if users type on the textbox and click "Post", a comment will be posted.

4.3.3 Functional Requirements

REQ-1: All participants of the project can comment it, not only the assignee;
REQ-2: Comment's author can edit them.

4.12 Adding/subtracting elapsed time to a task

4.5.1 Description and Priority

This feature allows to add or subtract time to the total spent on a specific task.

It has a medium priority, as it's not crucial for users, but it's definitely appreciated.

Priority components:

- Benefit: 7;
- Penalty: 8;
- Cost: 3;
- Risk: 2.

4.5.2 Stimulus/Response Sequences

On the task overview described in section 4.10, if users click the "+" button, they are prompted with a dialog. In this dialog they can insert time that will be added (or substracted) from the total elapsed time.

4.3.3 Functional Requirements

REQ-1: Users can either add or remove time. To remove time, they just need to put a "-" before the time specification;
REQ-2: The dialog must check if the time format is correct before confirming the action. If not, an error is prompted and the total elapsed time isn't affected;
REQ-3: Adding or removing time is always done on the total elapsed time of a specific task.

4.13 Task editing and deletion

4.5.1 Description and Priority

This feature allows to edit the details of a specific task or delete it.

It has a high priority, as users may need this function at any moment after creating a task.

Priority components:

- Benefit: 9;
- Penalty: 9;
- Cost: 2;
- Risk: 2.

4.5.2 Stimulus/Response Sequences

On the task overview described in section 4.10, if users click the "Edit task" button, they are prompted with a dialog. In this dialog they can edit all the fields described on section 4.9 and, also, delete the task by clicking "Delete task". If all the fields are valid, after clicking "Edit" the form is sent and the task edited accordingly.

4.3.3 Functional Requirements

REQ-1: When editing a task, users can specify: its title, its assignee (only one user), its time estimate, the due date and/or a description;

REQ-2: If the value of one or more fields is invalid, an error must be prompted and the task isn't created;

REQ-3: After deleting a task, users must be redirected to the project's Kanban board.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

There are some minimum recommended hardware requirements for a machine running TaskEase:

- 64-bits 1Ghz CPU;
- 2 GB RAM.

Please note that running TaskEase directly from the source code will require considerably less resources than the Docker deployment. Also, these are just recommended requirements, meaning TaskEase could also run on less capable machines, but with some performance degradation for final users. The internal memory requirements are not specified, as they may differ considerably on different implementations, depending strongly on the number of expected users.

5.2 Safety Requirements

TaskEase developers and collaborators cannot be considered responsible in the case of loss of information or for any damage that may be caused by bugs or incorrect use. Users should regularly backup their data when using TaskEase and not rely completely on it, as it’s an early-stage project with many flaws.

5.3 Security Requirements

It’s important to develop TaskEase in order to protect users’ information. Authentication and registration must be provided with state-of-the-art tools, designed to guarantee privacy and security. Note that the data saved by a user isn’t protected by design, but saved clear on the deployment’s DB. When deploying TaskEase publicly it’s important to note any privacy regulation to follow, such as the GDPR in the European Union.

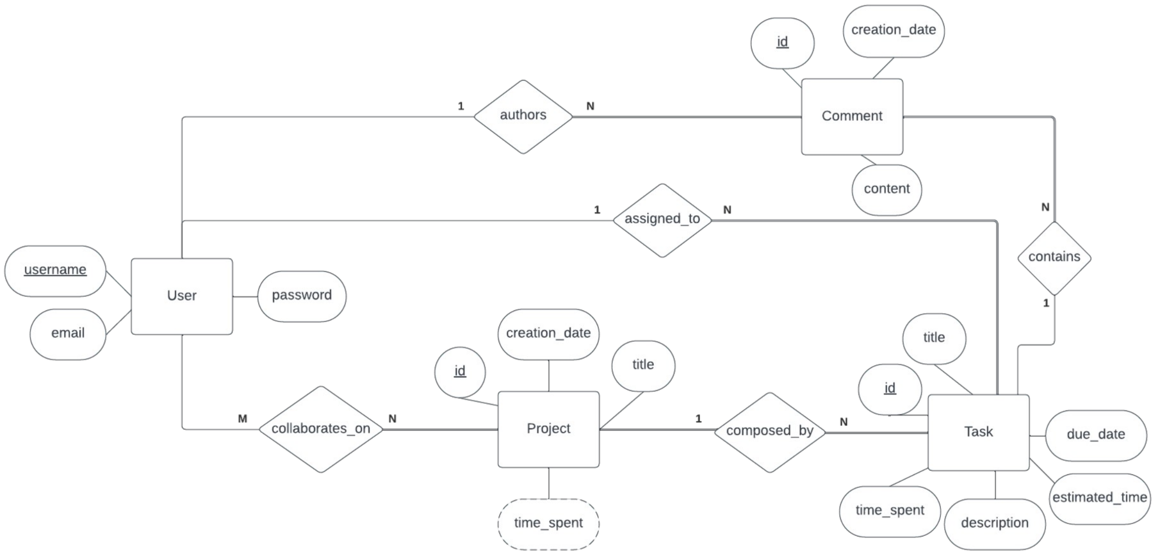
5.4 Software Quality Attributes

TaskEase software must be interoperable and portable, attributes guaranteed by the containerization with Docker. It also has to be easy to use by both developers and final users and at least its backend must be tested extensively to avoid errors and bugs. Finally the coding style should allow for an easy maintainability of the project.

5.5 Business Rules

All the development of TaskEase is made by Alessio Sordo (@alesordo on Github). He is the only person responsible for the management of this project at the present moment.

Appendix A: Analysis Models



Entity-Relationship Diagram